REMARKS

Claims 1 - 46 are pending. By this amendment, the Title, specification, and claims 1 - 4, 6 - 10, 12, 16 - 27, 30 - 33, 35, 36, 38 - 41, 45, and 46 are amended. No new matter is added. Reconsideration and issuance of a Notice of Allowance are respectfully requested.

TITLE

The Title is amended to more concisely relate to the claimed inventions and to conform to the claim amendments contained herein.

SPECIFICATION

On page 2 the Office Action objects to the specification. The specification is amended. Withdrawal of the objection to the specification is respectfully requested.

REJECTION OF THE CLAIMS UNDER 35 U.S.C. §112, ¶2

On page 2 the Office Action rejects claims 1 - 3, 5 - 9. 16, 21, 24, 25, 27, 30, 33, 35, 36, and 38 - 40 under 35 U.S.C. §112, ¶2. This rejection is respectfully traversed.

Considering claim 1, the Office Action states that the claim terms "on the fly", "orders", "components of an operating system" and "real dialog" are vague and indefinite.

Claim 1 is hereby amended to delete the term "on the fly" and to replace the terms "orders" with "location" and "sequence", and "real dialogue" with "communicates." Use of the tem "communicates" is supported in the specification at least at page 7, lines 17 - 22. Use of the term "orders" in claim 1 refers to the loading and execution of operating system components. Use of the term "sequence" in claim 1 as a replacement for "orders" is supported in the specification at least at page 8 line 29 - page 9, line 8: the sequence (order in which executed) of read and write commands is adjusted as needed to ensure "that the emulated hard disk is always coherent." However, the loading and execution of the operating system components may also be changed with respect to the "location" from which reads are made and to which writes are made. See, e.g., page 5, lines 2 - 21. Thus, the term "orders" in claim 1 is replaced with the terms "sequence" and "location." Concerning the term "on the fly" Applicants note that this is a common term that implies an operation is executed without an intermediate step; for example, storage of data to long term memory with out caching. In fact, the specification defines the term "on the fly" to mean "on the spur of the moment." See page 8, lines 16 - 17.

Considering the term "components of the operating system" Applicants note that operating system (OS) components, as a term, is well known to those skilled in the art. The basic components of an OS include:

- Process Management: Create/Delete/Suspend/Resume Processes, Inter Process Communication (IPC)
- Memory Management: Allocate and Deallocate Physical Memory
- Virtual Memory Management: Maintain mapping from virtual to physical memory, decide on size of memory to allocate to processes, enforce replacement policy
- I/O Management: definition of Device Drivers providing a uniform interface to devices
- Secondary Storage Management: allocation and deallocation of blocks, recovery block management.
- File Management: naming, directories, creation, use, and deletion of files, locks, protection, backups, accounting.
- Network Services: support for various protocols and communication facilities.
- Command Language Interpreter(CLI): interface to operator and users that gives them the ability to give commands to the OS.
- Information Services: Show/Set commands
- Accounting Services: Quotas, usages, and costs.

Considering claim 2, the Office Action states that the terms "peripheral driver level" and "level of an optional hard disk server service" are vague and indefinite. Applicants disagree.

"Peripheral driver level", as a well-known term of art, refers to the level of a computing system at which device drivers operate to enable (instruct) various peripheral devices. Also, as is well know in the art, the term "level" in the context of a computer system, refers to the hierarchical structure of that system, including the software loaded onto, or fixed in hardware devices that compose the computer system. A device driver is a specific type of computer software developed to allow interaction with hardware devices. Typically this constitutes an interface for communicating with the device, through the specific computer bus or communications subsystem that the hardware is connected to, providing commands to and/or receiving data from the device, and on the other end, the requisite interfaces to the operating system and software applications. It is a specialized hardware-dependent computer program which is also operating system specific that enables another program, typically an operating system or applications software package or computer program running under the operating system kernel, to interact transparently with a hardware device, and usually provides the requisite interrupt handling necessary for any necessary asynchronous time-dependent hardware interfacing needs. The key design goal of device drivers is abstraction. Every model of hardware (even within the same class of device) is different. Newer models also are released by manufacturers that provide more reliable or better performance and these newer models are often controlled differently. Computers and their operating systems cannot be expected to know how to control every device, both now and in the future. To solve this problem, operating systems essentially dictate how every type of device should be controlled. The function of the device driver is then to translate these OS mandated function calls into device specific calls. In theory, a new device, which is controlled in a new manner, should function correctly if a suitable driver is available. This new driver will ensure that the device appears to operate as usual from the operating systems' point of view for any person.

Also as is well known in the art, a server service refers to a set of features or capabilities that are provided by a central computing platform (a server, or host) to a number of subordinate client computers, processors, or software applications. A "level of an optional server service", therefore, refers to a higher level than the driver level, for example, and more specifically refers to the level of the server that provides the features or capabilities of the server to the client computers, processors, and applications.

Claim 3 is rejected for some of the same reasons as applied to claim 2; just as with claim 2, Applicants contend that claim 3 is not vague and indefinite.

Considering claim 5, the amendment to claim 1 corrects the stated lack of antecedent basis.

Considering claim 6, the Office Action states that the term "low level micro software module" is vague and indefinite, and that the claim as a whole is vague and indefinite. Claim 6 is amended to overcome this rejection.

Claim 7 is amended to overcome the stated rejection based on vagueness, indefiniteness, and lack of antecedent basis.

Claims 8, 9, and 16 are rejected as containing vague and indefinite terms. Claims 8, 9, and 16 are amended to overcome these rejections.

Considering claim 21, the Office Action states that the term "wherein a specific program called 'server software'" is vague and indefinite. Claim 21 is amended to recite a "server program." Support for this amendment can be found at least at page 6, lines 10 - 11.

The remaining claims 24, 25, 27, 30, 33, 35, 36, 38, 39, and 40 are rejected as containing terms that are vague and indefinite, or lack antecedent basis. These claims are amended to overcome the rejections. In addition, other claims are amended to conform with the above described claim amendments, and to clarify their subject matter.

In view of the above remarks and the claim amendments, Applicants respectfully assert that each claim now meets all the requirements of 35 U.S.C. §112, ¶2, and hence Applicants request withdrawal of the rejections of claims 1 - 3, 5 - 9. 16, 21, 24, 25, 27, 30, 33, 35, 36, and 38 - 40 under 35 U.S.C. §112, ¶2.

REJECTION OF THE CLAIMS UNDER 35 U.S.C. §101

On page 4 the Office Action rejects claims 1 - 46 under 35 U.S.C. §101 as directed to non-statutory subject matter. This rejection is respectfully traversed.

Claim 1 is amended to conform to the requirements of 35 U.S.C. §101. Specifically, claim 1 is amended to recite that the method is performed by a suitably programmed processor and that the method transforms programming contained on a hardware disk into a device capable of controlling read and write operations on a client station and among two or more client stations. Support for this amendment can be found in the specification at least at page 5, line 14 - page 6, line 21.

REJECTION OF THE CLAIMS UNDER 35 U.S.C. §102(b) AND §103(a)

On page 5 the Office Action rejects claims 1 - 6, 11 - 29, and 38 - 46 under 35 U.S.C. 102(b) over Flouris: "The Network RamDisk: Using Remote Memory on

Heterogeneous NOWs," (hereafter Flouris). On page 13, the Office Action rejects claims 7 - 10 and 30 - 37 under 35 U.S.C. 103(a) over Flouris in view of U.S. Patent Publication 20030208675 to Burokas (hereafter Buorkas). These rejections are respectfully traversed.

Claim 1

The Office Action asserts that Flouris discloses "a method for software emulation of hard disks ... with parameterizable management ... of requests for writing and reading data ... ." More particularly, the Office Action asserts that Flouris, at page 10, when stating that a "NRD client asks [sic] as a normal disk" anticipates the claimed "creating a representation of a real hard disk." Applicants respectfully disagree with this characterization of Flouris.

Flouris is directed to a disk driver that is intended to reduce latency of data storage operations (reads and writes) by using idle main memory of two or more networked computers or workstations rather than using "traditional magnetic disks." See Flouris, page 2. Flouris accomplishes this reduction in latency (as much as a factor of ten - see page 2) by running a NRD server process on each workstation, where the NRD server process searches for idle main memory blocks among the networked workstations. These idle (i.e., available) main memory blocks are listed in a "block table" that is accessible by the NRD server processes. Nowhere does Flouris disclose or suggest creating a representation of an actual (real) hard disk.

In contrast to Flouris, claim 1 recites a method for software emulation of hard disks including the step of creating a representation of a real hard disk, wherein the sequence and location for loading and execution of components of an operating system may be modified. Accordingly, claim 1 is patentable.

Claims 2 - 6, 11 - 29, and 38 - 46

Claims 2 - 6, 11 - 29, and 38 - 46 depend from patentable claim 1. For this reason and the additional features they recite, claims 2 - 6, 11 - 29, and 38 - 46 also are patentable.

Claims 7 - 10 and 30 - 37

Claims 7 - 10 and 30 - 37 depend from patentable claim 1. For this reason and the additional features they recite, claims 7 - 10 and 30 - 37 also are patentable.

In view of the above remarks and amendments, Applicants respectfully request withdrawal of the rejections of claims 1 - 46 under 35 U.S.C. §102(b) and §103(a).

Should the Examiner believe that anything further is desired in order to place the application in even better condition for allowance; the Examiner is invited to contact Applicants' undersigned representative at the telephone number listed below.

A three-month extension fee is believed to be due. The Examiner is hereby authorized to charge the fees to **Deposit Account No. 08-2025**, and notify the undersigned.

Respectfully submitted,

Date: March 26, 2009

John K. Harrop
Registration No. 41,817
ANDREWS KURTH LLP
1350 I Street, NW, Suite 1100
Washington, D.C. 20005
Telephone: (202) 662-3050
Fax: (202) 662-2739